AMENDMENTS TO THE CLAIMS

The following is a complete, marked-up listing of revised claims with a status identifier in parentheses, underlined text indicating insertions, and strikethrough and/or double-bracketed text indicating deletions.

LISTING OF CLAIMS

1. (currently amended) A method for controlling data flow using a leaky bucket data flow control ~~scheme~~ algorithm, the method comprising:

adjusting an amount of information transmitted to and from a leaky bucket using a granularity scalar parameter in a leaky bucket data full ratio of the leaky bucket data flow ~~scheme~~ algorithm ~~with a granularity scalar parameter~~, the leaky bucket data full ratio indicating a current amount of data in a leaky bucket in relation to a maximum data capacity of the leaky bucket, and the granularity scalar parameter modifying the indication of the leaky bucket data full ratio ~~flow control scheme to modify the granularity of information relating to the level of the leaky bucket to control data flow~~.

2. (original) The method according to claim 1, further comprising:
adjusting the scalar parameter based upon a user determined scaling value.

3. (original) The method according to claim 1, wherein the scalar parameter is within a predetermined range.

4. (original) The method according to claim 2, wherein the step of adjusting is performed dynamically.

5. (original) The method according to claim 1, wherein the scalar parameter modifies a bucket full ratio.

6. (currently amended) A method for data flow control comprising:

scaling a control parameter <u>in a leaky bucket data full ratio</u> of a leaky bucket data flow ~~scheme for~~ <u>algorithm to</u> adjust[[ing]] <u>an amount of information transmitted to and from a leaky bucket,</u> the <u>leaky bucket data full ratio indicating a current amount</u> ~~granularity~~ of information <u>in the leaky bucket in relation</u> ~~relating~~ to the <u>maximum</u> level of the leaky bucket ~~for controlling data flow~~, the control parameter modifying ~~a bucket capacity parameter for~~ the leaky bucket data <u>full ratio indication</u> ~~flow control scheme~~.

7. (original) The method according to claim 6, wherein the scaling is performed within a predetermined range.

8. (original) The method according to claim 7, wherein the predetermined range is between an empty bucket level and a maximum bucket level.

9. (original) The method according to claim 7, further comprising:
using a user defined scaling value for scaling the control parameter.

10. (original) The method according to claim 7, wherein the bucket capacity parameter is a bucket full ratio for the leaky bucket data flow scheme.

11. (currently amended) The method according to claim 7, further comprising:
dynamically adjusting ~~the granularity~~ <u>a number of data bits in a leaky bucket data full ratio</u> based upon scaling of the control parameter.

12. (original) The method according to claim 7, further comprising:
varying data flow based upon scaling of the control parameter.

13. (previously presented) A method for controlling data flow using a leaky bucket data flow control ~~scheme~~ <u>algorithm</u>, the method comprising:
modifying <u>a granularity scalar parameter in a leaky bucket data full ratio of the leaky bucket data flow control algorithm to adjust an indication of a leaky bucket current data level relative to a leaky</u> bucket [[a]] <u>maximum data, the indication</u> capacity

~~indicator~~ provided by the bucket data full ratio, thereby enhancing leaky bucket data flow as controlled by ~~to provide enhanced granularity of information relating to the level of the leaky bucket to enhance the granularity of~~ the leaky bucket data flow control ~~scheme~~ algorithm.

14. (original)   The method according to claim 13, wherein the bucket capacity indicator is a bucket full ratio.

15. (original)   The method according to claim 13, wherein the step of modifying comprises:
using a scalar value to modify the bucket capacity indicator.

16. (original)   The method according to claim 15, further comprising:
establishing the scalar value based upon system requirements.

17. (original)   The method according to claim 15, further comprising:
dynamically changing the scalar value.